Confidential Computing within an AI Accelerator

Kapil Vaswani, **Stavros Volos,** Cédric Fournet, Antonio Diaz, Ken Gordon, Balaji Vembu, Sam Webster, David Chisnall, Saurabh Kulkarni, Graham Cunningham, Richard Osborne, Daniel Wilkinson

Microsoft and Graphcore

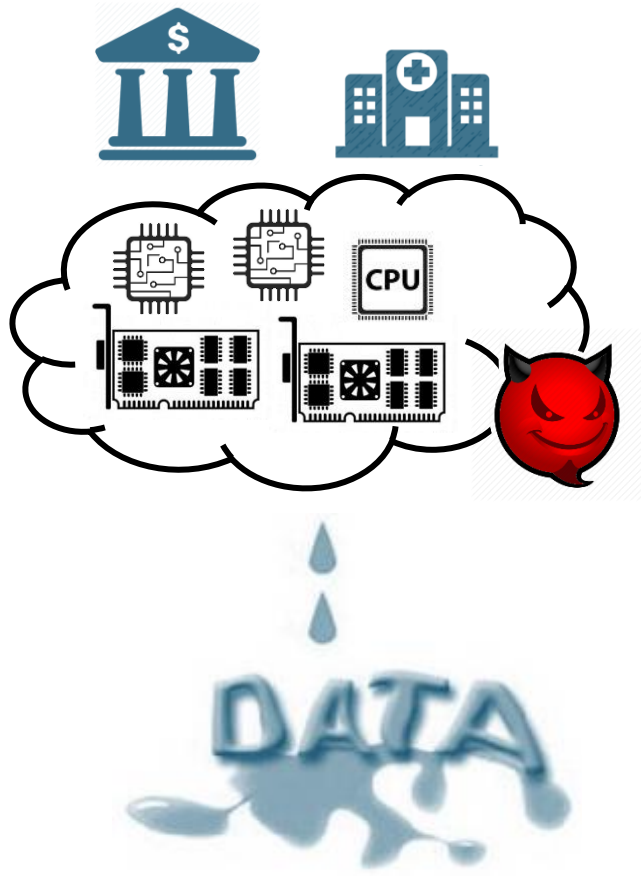
Conflicting Cloud Trends: AI & Privacy

Cloud Accelerators Enable Hyper-scale AI

- Medical diagnostics, financial forecasting, generative AI
- Large models require 10^N high-end GPUs

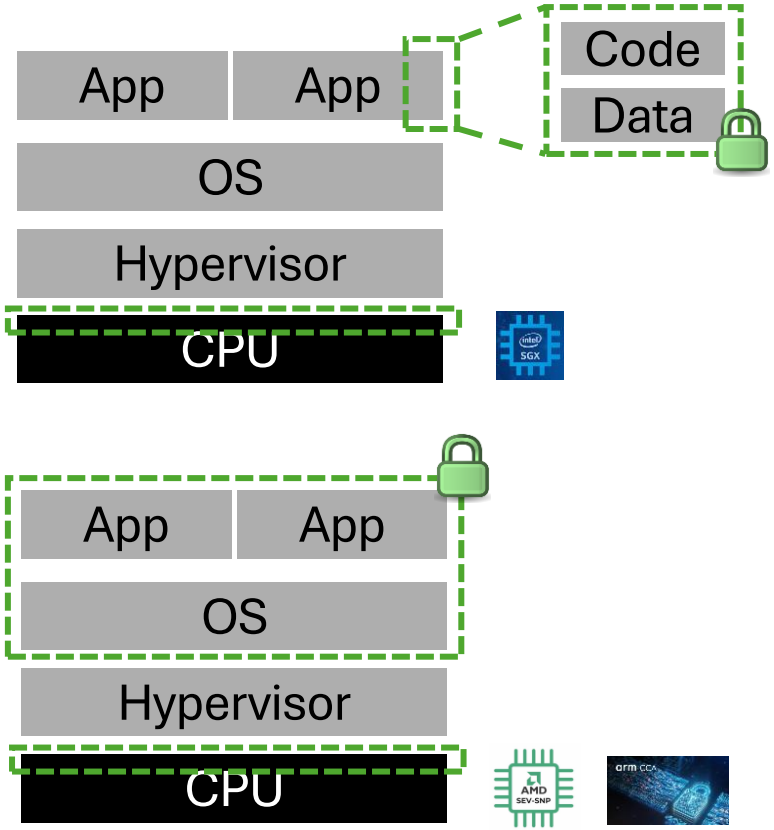
Ever-growing Confidentiality & Privacy Concerns

- Privacy-sensitive data (e.g., medical history, transactions)
- Proprietary AI models
- Sharing of infrastructure, sophisticated attacks



Need strong security mechanisms for preserving cloud AI privacy

Confidential Computing to the Rescue



Trusted Execution Environments (TEE)

- Execution isolated from privileged attackers
- Remote attestation for establishing trust
- Support by major CPU vendors (past 10 years)
  - Process-based vs. VM-based

Limited Support in PCIe Devices

- Research proposals, e.g., Graviton [OSDI'18]
- Upcoming NVIDIA Hopper GPUs
  - Works in conjunction with VM-based TEEs

→ Large TCB, dependency with host CPU

Need low-TCB and flexible AI Accelerator TEE

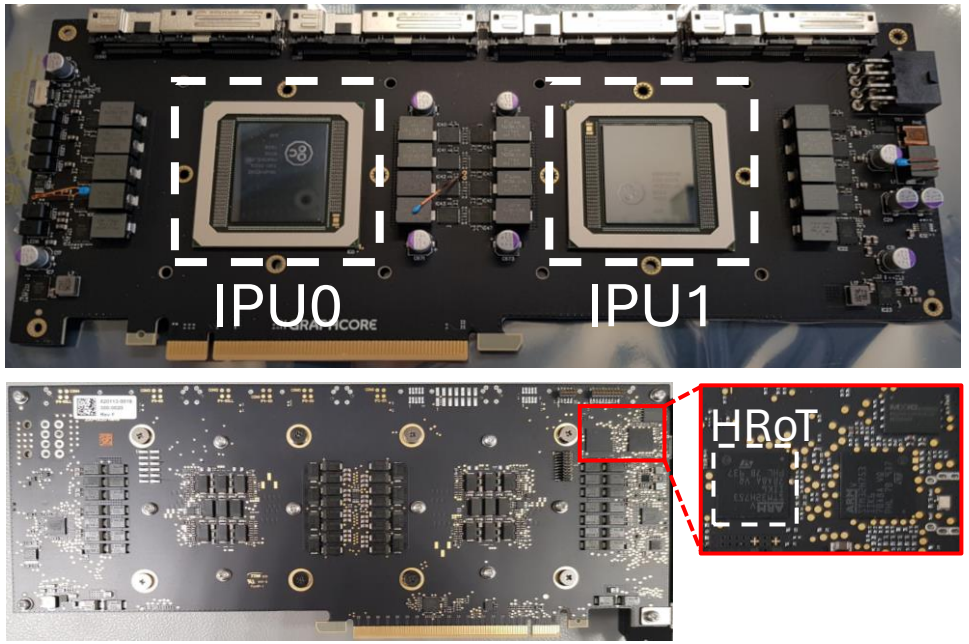
ITX: A Case for a Confidential AI Accelerator

Security Guarantees

- Confidentiality, integrity of computation & data
- Remote attestation

IPU Trusted Extensions (ITX)

- No trust in CPU
- No changes to programming model
- Low performance overheads



Development board manufactured in 2020

Outline

- Introduction
- Graphcore IPU and Threat Model
- IPU Trusted Extensions (ITX)
- Conclusion

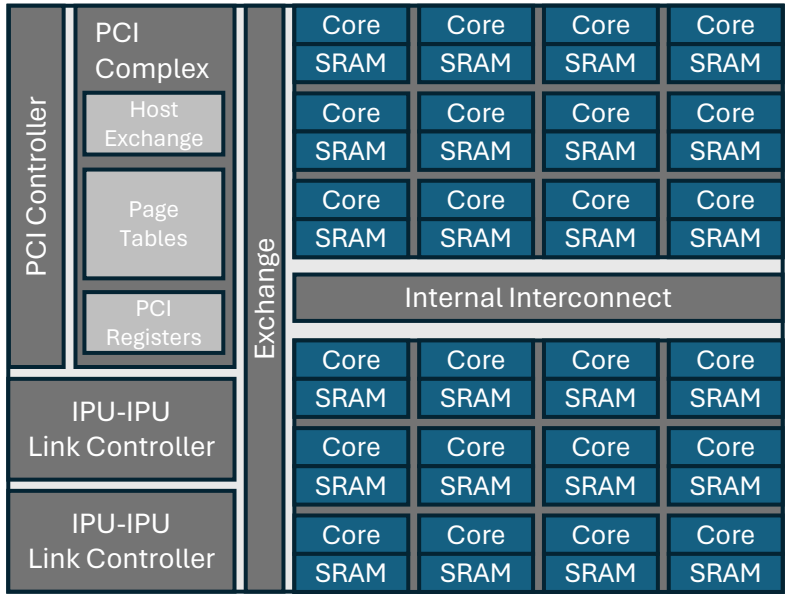
IPU Hardware Architecture

Tiled Architecture (>1400 tiles)

- Compute: in-order multi-threaded core with AI units
- Memory: fixed-latency SRAM
  - No caching, mapped to tile's address space

PCI Complex

- Mediates IPU-Host communication
- Host memory mapped to tiles via page tables
- IPU memory mapped to host via PCI BAR



How do we program such a massively parallel processor?

IPU Execution Model

Bulk Synchronous Parallel Paradigm

- Compute and exchange phases
- Internal synchronization for Tile-Tile I/O
- External synchronization for Host-Tile I/O

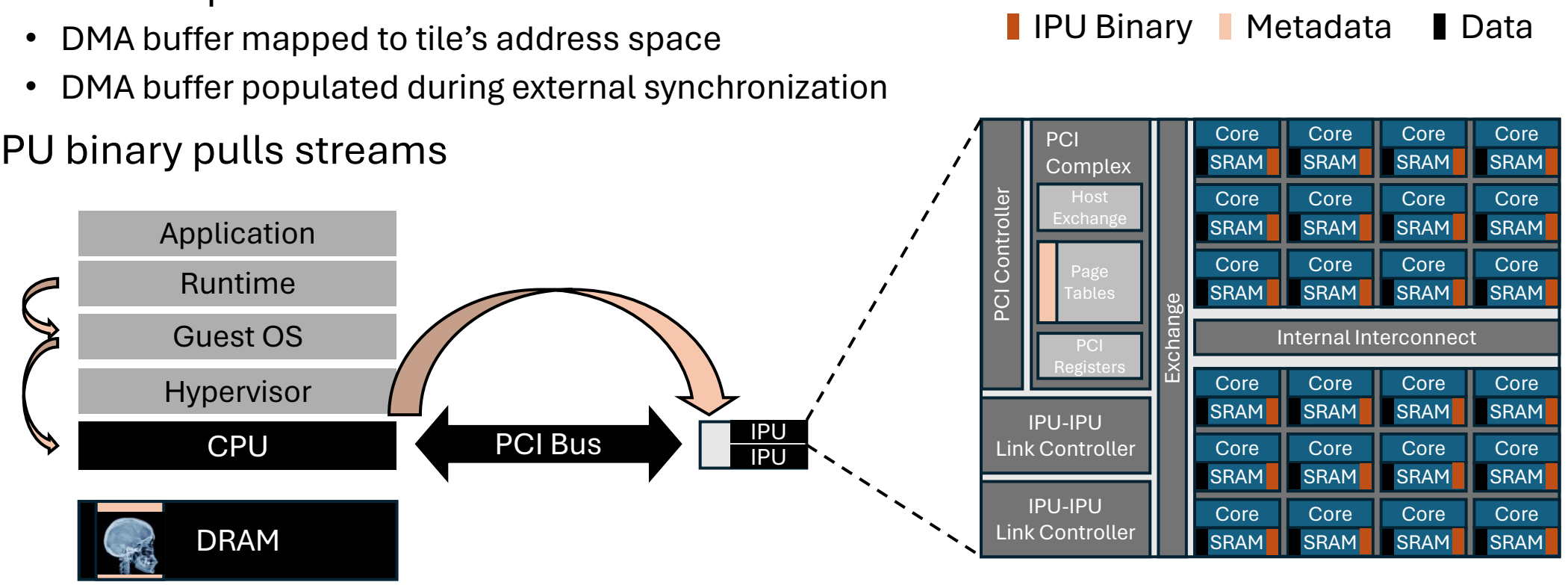
Compiler-driven Resource Allocation

- Each node in the ML graph is assigned tile resources (threads, memory)
- IPU binary defines its entire control and data flow
  - Compute and exchange phases defined at compilation time

How does the IPU software stack support external I/O?

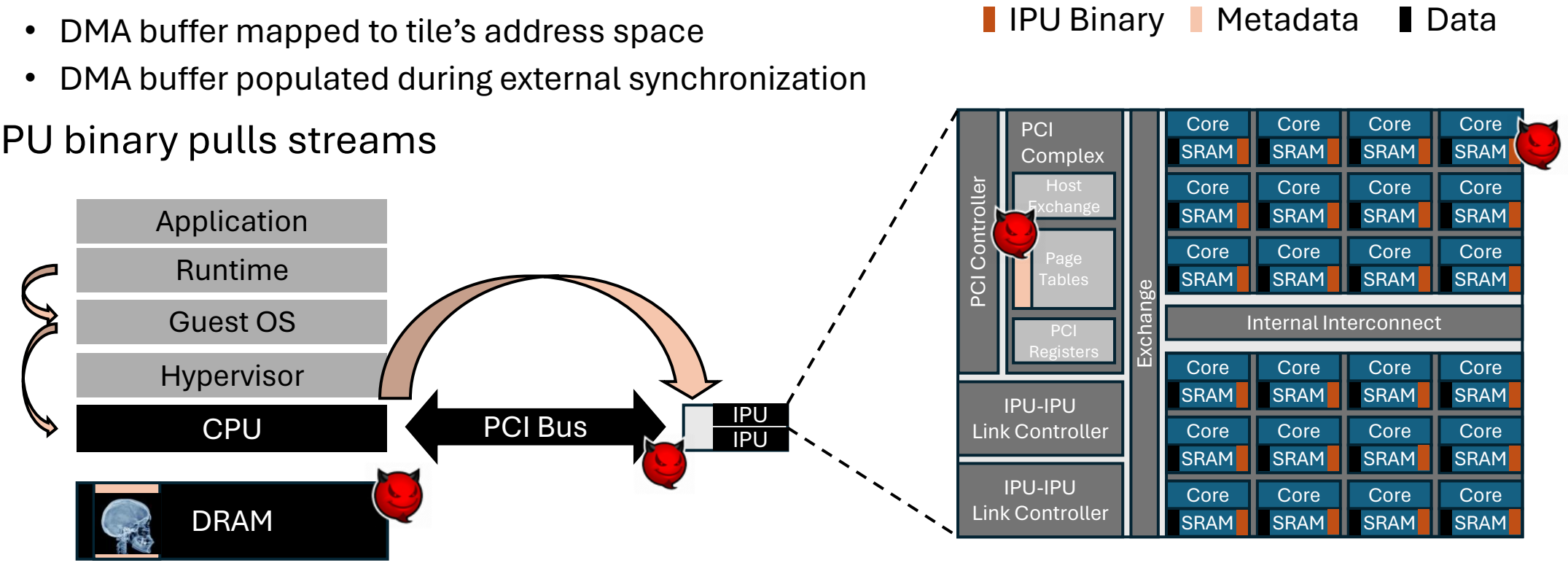
Streams: An External I/O Abstraction

- Compiler maps external I/O to streams, generates tile code for DMA
- Runtime implements streams
  - DMA buffer mapped to tile's address space
  - DMA buffer populated during external synchronization
- IPU binary pulls streams



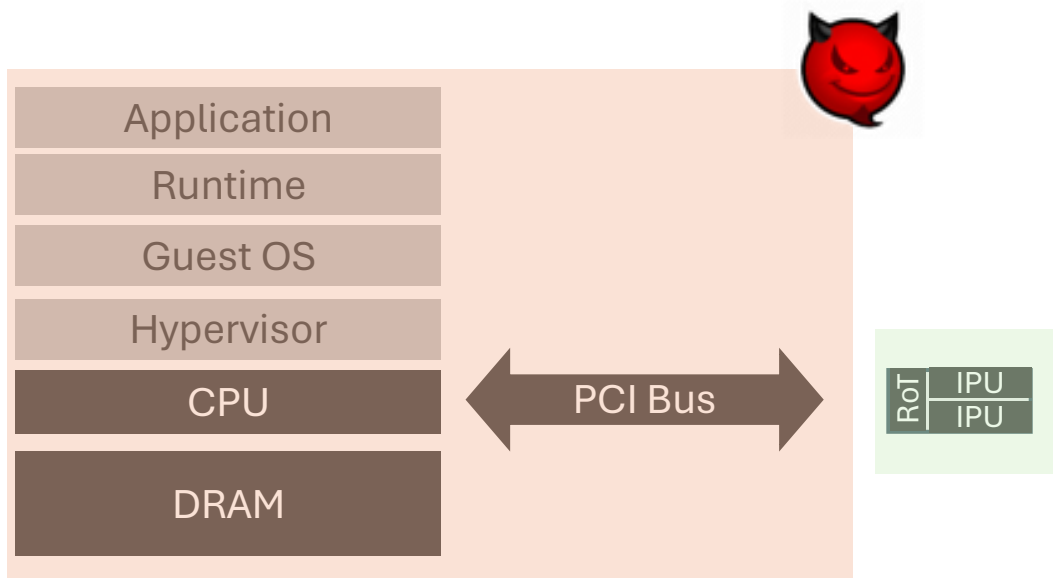
Streams: An External I/O Abstraction

- Compiler maps external I/O to streams, generates tile code for DMA
- Runtime implements streams
  - DMA buffer mapped to tile's address space
  - DMA buffer populated during external synchronization
- IPU binary pulls streams



The external I/O path has a large attack surface

Threat Model



Trusted Computing Base (TCB)

- IPU Package including on-chip SRAM
- IPU HRoT (HW and FW)
- ML Framework and IPU Compiler

Goal: Confidentiality and integrity of computation and data

Out-of-scope

- Physical side-channel attacks (e.g., DPA)
- Package manufacturing attacks

Outline

- Introduction
- Graphcore IPU and Threat Model
- IPU Trusted Extensions (ITX)
- Conclusion

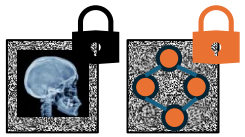
ITX in a Nutshell

- Decoupled compilation from execution
- Application binary & data can be prepared at tenant's premises
- Runtime mediates data using a compiler-generated job manifest

Trusted Mode
- Set by CCU during TEE initialization
- No sharing in space or time
- Blocks the host from IPU access

AES units for encrypting & authenticating communication over PCIe

Confidential Compute Unit responsible for code attestation and key provisioning



sync0
syncN

Application
Runtime
Guest OS
Hypervisor
CPU

DRAM

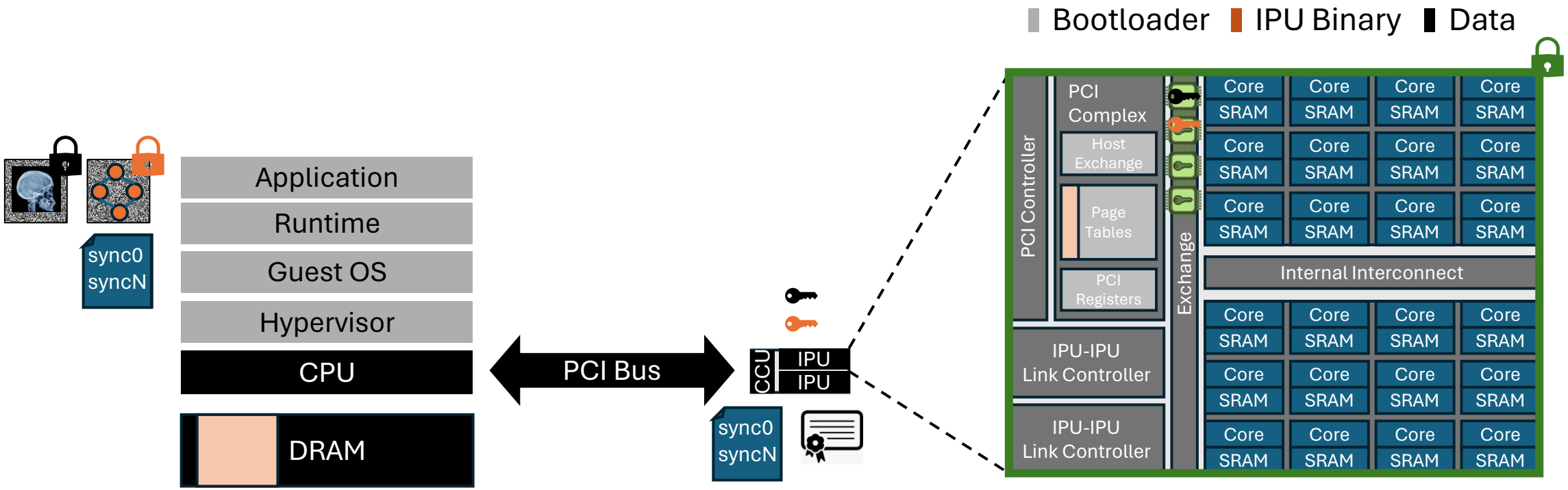
PCI Bus

CCU
IPU
IPU

PCI Controller
PCI Complex
Host Exchange
Page Tables
PCI Registers
IPU-IPU Link Controller
IPU-IPU Link Controller
Exchange
Internal Interconnect
Core
SRAM

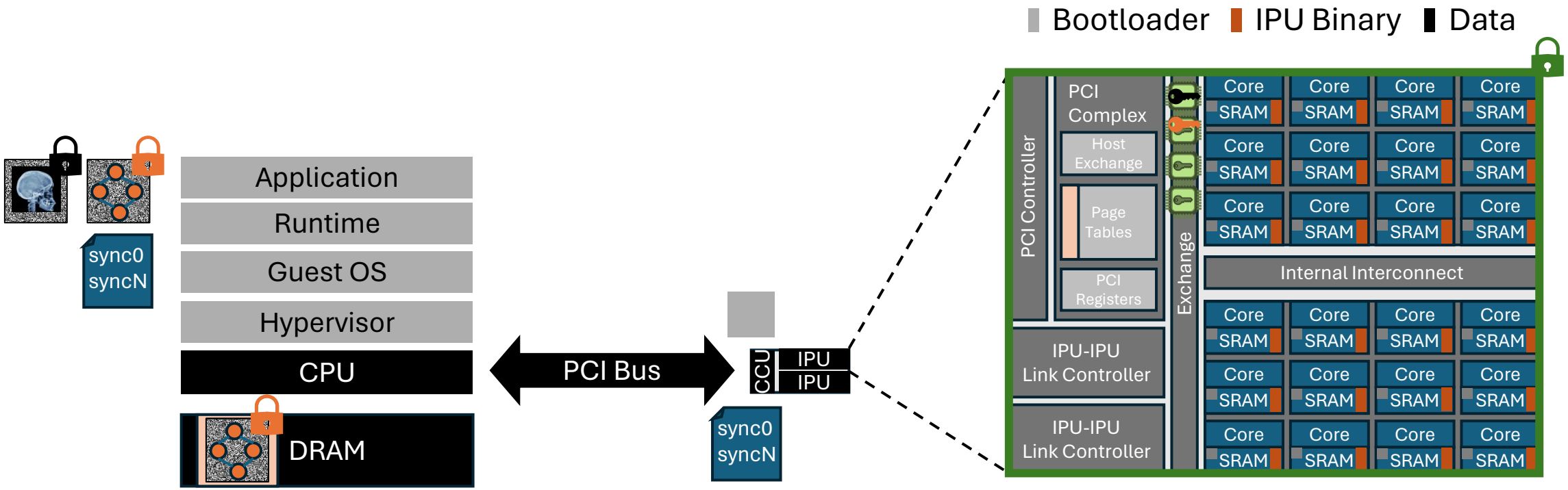
ITX in a Nutshell: Attestation & Key Provisioning

- CCU receives job manifest, issues attestation, receives and deploys keys from verifier(s)



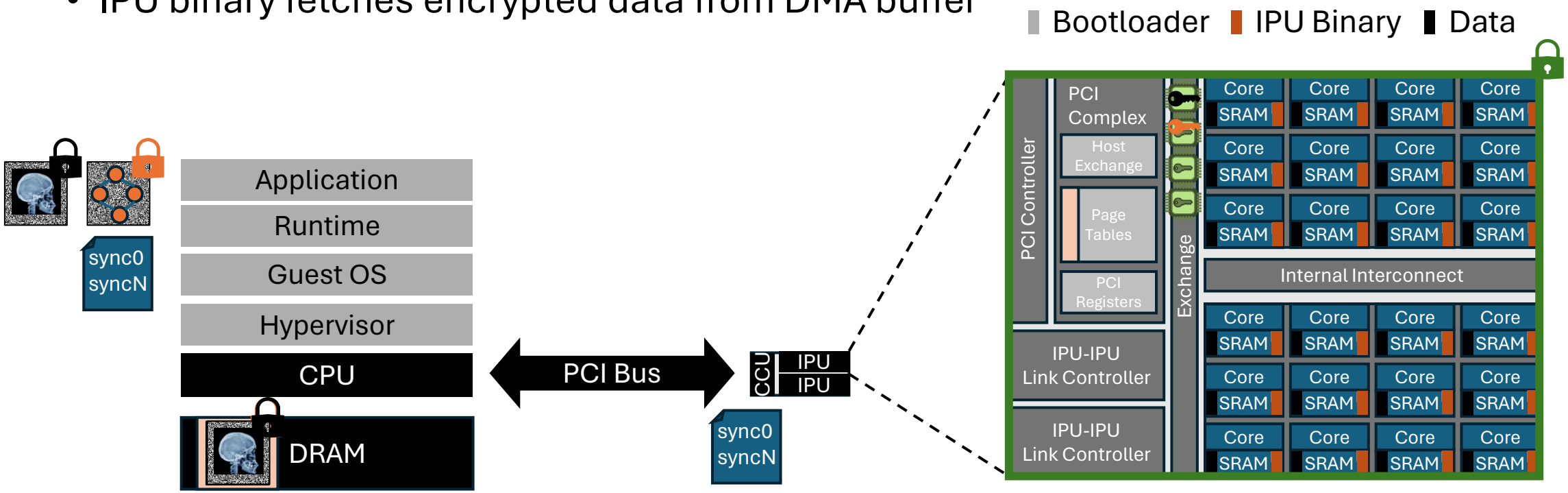
ITX in a Nutshell: Secure Bootstrapping

- CCU receives job manifest, issues attestation, receives and deploys keys from verifier(s)
- CCU deploys bootloader to load encrypted IPU binary, computes hash of the IPU binary



ITX in a Nutshell: Confidential Streams

- CCU receives job manifest, issues attestation, receives and deploys keys from verifier(s)
- CCU deploys bootloader to load encrypted IPU binary, computes hash of the IPU binary
- IPU binary fetches encrypted data from DMA buffer



Need to prevent reorder and replay attacks from host CPU

Encrypted Direct Memory Access

- Compiler maps all streams into a stream address space (AS)
- Compiler partitions the application's streams into frames
- Frames identified with a unique ASID acting as an IV during encryption
  - IV enclosed within frame for efficient supply to the HW decryption
- HW decryption ensures integrity based on tag
- The receiver verifies that the frame's IV matches the requested ASID

Low-cost protection against replay and re-order attacks

More Details In the Paper

- Measured boot protocol
- Firmware provisioning with no trust in supply chain
- Firmware updates w/o device re-certification

- TEE lifecycle (initialization, launch, termination)
- Remote attestation protocol

- Multi-key support in encryption engines
- Secure checkpointing

In Proceedings of the 2023 USENIX Annual Technical Conference (ATC'23)

Confidential Computing within an AI Accelerator

Kapil Vaswani[1], Stavros Volos[1], Cédric Fournet[1]
Antonio Nino Diaz[1], Ken Gordon[1], Balaji Vembu[3,†], Sam Webster[1], David Chisnall[1], Saurabh Kulkarni[4,†]
Graham Cunningham[5,‡], Richard Osborne[2], Daniel Wilkinson[6,‡]

[1]*Microsoft* [2]*Graphcore* [3]*Meta* [4]*Lucata Systems* [5]*XTX Markets* [6]*Imagination Technologies*

Abstract

We present IPU Trusted Extensions (ITX), a set of hardware extensions that enables trusted execution environments in Graphcore's AI accelerators. ITX enables the execution of AI workloads with strong confidentiality and integrity guarantees at low performance overheads. ITX isolates workloads from untrusted hosts, and ensures their data and models remain encrypted at all times except within the accelerator's chip. ITX includes a hardware root-of-trust that provides attestation capabilities and orchestrates trusted execution, and on-chip programmable cryptographic engines for authenticated encryption of code/data at PCIe bandwidth.

We also present software for ITX in the form of compiler and runtime extensions that support multi-party training without requiring a CPU-based TEE.

We included experimental support for ITX in Graphcore's GC200 IPU taped out at TSMC's 7nm node. Its evaluation on a development board using standard DNN training workloads suggests that ITX adds $< 5\%$ performance overhead and delivers up to 17x better performance compared to CPU-based confidential computing systems based on AMD SEV-SNP.

1 Introduction

Machine learning (ML) is transforming many tasks such as medical diagnostics, video analytics, and financial forecasting. Their progress is largely driven by the computational capabilities and large memory bandwidth of AI accelerators such as NVIDIA GPUs, Alibaba's NPU [2], Google's TPU [18], and Amazon's Inferentia [3]. Their security and privacy is a serious concern: due to the nature and volume of data required to train sophisticated models, the sharing of accelerators in public clouds to reduce cost, and the increasing frequency and severity of data breaches, there is a realization that machine learning systems require stronger end-to-end protection mechanisms for their sensitive models and data.

[†]Work done while at Microsoft; [‡]Work done while at Graphcore.

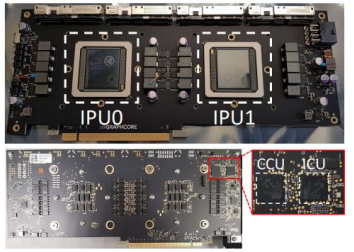


Figure 1: Graphcore Intelligence Processing Unit (IPU) development board (May 2020) with ITX extensions, showing two IPUs on the front side connected to the CCU via the ICU on the back.

Confidential computing [1, 4, 11, 31] relies on custom hardware support for trusted execution environments (TEE), also known as enclaves, that can provide such security guarantees. Abstractly, a TEE is capable of hosting code and data while protecting them from privileged attackers. The hardware can also measure this code and data to issue an *attestation report*, which can be verified by any remote party to establish trust in the TEE. In principle, confidential computing enables multiple organizations to collaborate and train models using sensitive data, and to serve these models with assurance that their data and models remain protected. However, the predominant TEEs such as Intel SGX [22], AMD SEV-SNP [5], Intel TDX [16], and ARM CCA [6] are limited to CPUs. Recently, NVIDIA has announced TEE support in upcoming Hopper GPUs [25] that works in conjunction with CPU TEEs.

Adding native support for confidential computing into AI accelerators can greatly increase their security, but also involves many challenges. Security features such as isolation, attestation, and side-channel resilience must be fitted in their

Experimental Setup

IPU development board, manufactured at TSMC's 7nm

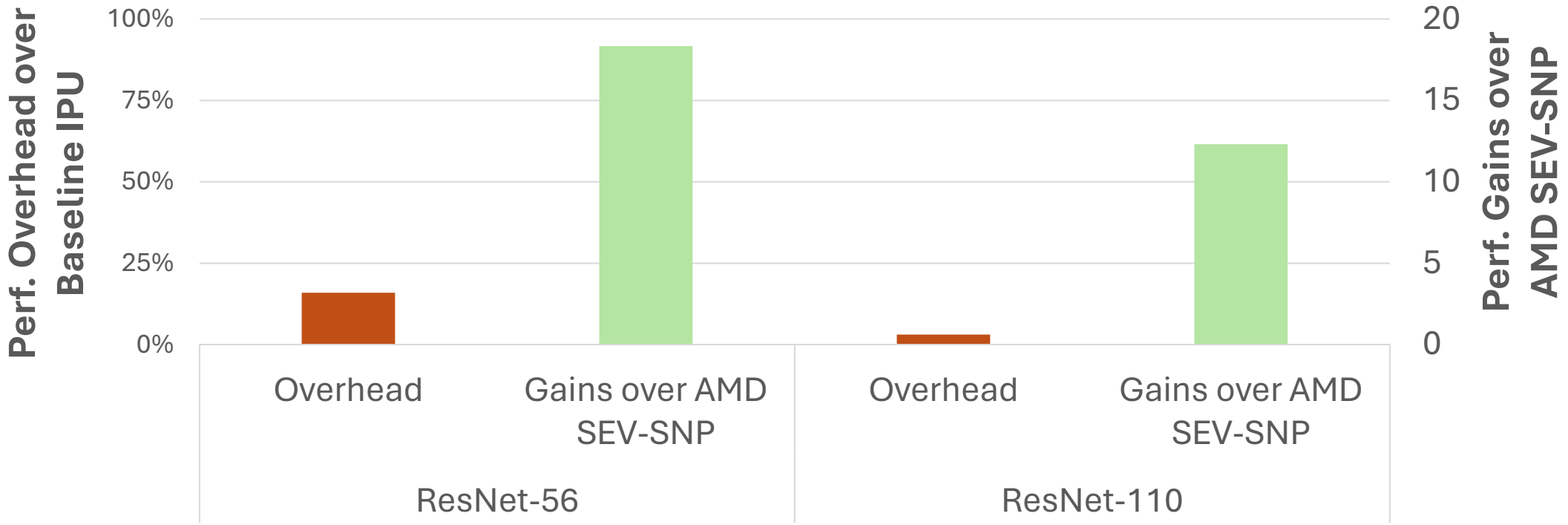
- Engineering samples operating at 900 MHz (of possible 1400MHz)

ResNet-X Training with Cifar-10 dataset

- At the time of evaluation, ResNet was a key ML workload

Comparison against VM-based TEE on 48-core AMD Zen3 Azure VM

Experimental Results



- For small models, ITX overheads dominated by TEE creation, key deployment
- For large models, overheads are amortized, accounting for ~3%
- Single ITX-enabled IPU delivers 12-18x higher performance than CPU TEEs

Conclusion

Ever-growing need for hyper-scale privacy-preserving AI

ITX: First ASIC enabling high-performance confidential AI

- Strong security guarantees
- Low TCB without trust on the CPU
- Low performance overheads

Since then, we have seen support by the AI ecosystem (NVIDIA)

Confidential Computing within an AI Accelerator

Stavros Volos (svolos@microsoft.com)

Azure Research, Microsoft